Lastni prostori panoramskih slik za samolokalizacijo mobilnega robota

Eigenspaces of panoramic images for mobile robot self–localization

Matjaž Jogan

Magistrska naloga
predložena
Fakulteti za računalništvo in informatiko
Univerze v Ljubljani
kot delna izpolnitev pogoja za pridobitev naslova
magister računalništva in informatike

April 2002

Mentor:
prof. dr. Aleš Leonardis

Magistrska naloga je bila izdelana pod mentorstvom prof. dr. Aleša Leonardisa in je last Fakultete za računalništvo in informatiko v Ljubljani. Za objavljanje in uporabo rezultatov diplomskega dela je potrebno soglasje zgoraj omenjene ustanove.

Tekst je oblikovan z LaTeX urejevalnikom besedil.

Abstract

Appearance–based visual learning and recognition techniques that are based on models derived from a learning set of 2D images are being widely used in computer vision applications. In robotics, they have received most attention in visual servoing and navigation. In this paper we discuss a framework for visual self–localization of mobile robots using a parametric model built from panoramic snapshots of the environment. Particularly, we propose solutions to the problems related to robustness against occlusions and invariance to the rotation of the sensor. Our principal contribution is an "*eigenspace of spinning–images*", i.e., a model of environment which successfully exploits some of the specific properties of panoramic images in order to efficiently calculate the optimal subspace in terms of Principal Components Analysis (PCA) of a set of training snapshots without actually decomposing the covariance matrix. By integrating a robust recover–and–select algorithm for the computation of image parameters we achieve reliability even in the case when the input images are partly occluded or noisy. In this way, the robot is capable of localizing itself in realistic environments.

Povzetek

Problem lokalizacije in orientacije v prostoru je dandanes aktualen predvsem zaradi potrebe po samolokalizaciji avtonomnih mobilnih robotov, pa tudi na področju virtualne in obogatene resničnosti.

Contents

List of Figures

Chapter 1

Introduction

With the rapid development in the field of computer vision, more and more robot systems use cameras that provide them with pictorial information about their surrounding. Compared to other sensors like sonar, laser beams or infrared cells, cameras are conveying a much richer visual information on the world around us, providing us with cues about texture, shape and illumination. However, the task of interpreting this information in a useful way is very complex.

Self–localization is one of the crucial capabilities of a mobile robot that enables autonomous navigation in the environment. While estimating the distance to some object can be efficiently performed by sonars or laser beams, visual systems were designed that can construct a 3D model of the surrounding by using only the visual input, and then use that model for mobile robot self–localization. Recent examples of such systems are Se et *al.* [25] that use scale-invariant image features as landmarks to build a 3D map and estimate robot ego-motion by matching them, or Murray et *al.* [19], who use a stereo camera rig to produce a 2D occupancy map.

Over the last decade, several researchers have questioned, whether models obtained by a 3D reconstruction of the environment are really indispensable for an artificial vision system. By following the findings of psychologists, more and more theories awoke that describe the visual process as a task of recognizing and associating 2D images. In computer vision, this resulted in a series of successful algorithms that use 2D images of objects to build a representation based solely on the appearance itself [20, 21, 18, 15]. In practice, this means that the representation does not contain explicit information on 3D metrics, but is rather constructed in a way that allows for view recognition using algorithms that have roots in statistical analysis of patterns.

Such appearance–based models are expected to have a plethora of advantages over the 3D models. Instructive collections of work that stresses the relation between the 3D reconstruction approach and the appearance–based visual recognition can be found in [24] and [1].

In robotics, appearance–based learning and recognition techniques have received most attention in visual servoing and self–localization, that can both be used in autonomous navigation. By defining the task of self-localization as the problem of recognizing the momentary view of the environment and then associating it with the position of the robot in the physical space, it is evident that appearance–based models can be used. This brings several advantages — firstly, there is no need for intensive preprocessing of the visual input, as there is no need to search for prominent features and their correspondences. On the contrary, when using an appearance–based model, the image can be used without extracting local features, as visual information gets interpreted in a rudimentary form. Furthermore, 3D models are hard to build for complex environments.

Several research groups have presented successful methods for appearance–based localization that differ mostly in the type of optical sensors and in the way the models are built [9, 2, 22, 7, 13, 11]. We will give a brief overview of some of these approaches in Section 2.

Additional motivation for this research comes from the discoveries on navigation strategies of insects, that are, although limited in the brain size, capable of amazingly confident self-localization. In fact, some studies (see [14] and the references therein) imply, that wood ants may be using a representation of the environment that is built from wide-angle snapshots of the scene. Localization is then performed by comparing the instantaneous view with the stored snapshots. According to this and some other studies, e.g. Dill [6], the patterns are processed retinotopically, i.e., the snapshot is not segmented, but interpreted as a whole.

The notion that ants can localize themselves using methods that resemble appearance matching although their processing resources being limited is a sufficient reason for looking further in solutions that nature itself is serving us. The next useful hint is that having a wide field–of–view can be helpful for localization. By using a catadioptric camera [8] which acquires panoramic images, we can extend the robot's field–of–view far beyond that provided by classical cameras. Furthermore, we can expect that such views taken from nearby positions and oriented in the same direction will be strongly correlated, as the visual information contained in the panoramic images, does not change so abruptly as it does on the image-plane of standard cameras with a limited field-of-view. This allows us not only to design an efficient strat-

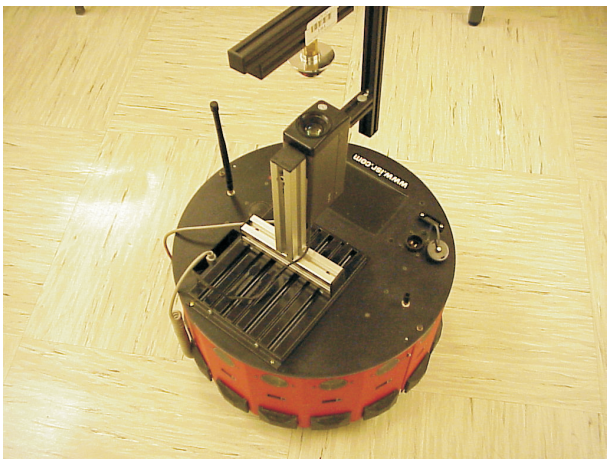


Figure 1.1: Mobile robot with a mounted catadioptric panoramic sensor.

egy based on the distance in the image space, but also to build a compact representation that eliminates redundancy.

In our work we describe a method for localization of a mobile system which is based on recognizing a panoramic view using a model built from 2D panoramic images acquired in the learning phase. The model is constructed by using an approximation of the image set, which gets represented by a low-dimensional set of principal components or *eigenvectors* that span a low-dimensional *eigenspace*. Projections of images on this subspace are the basis for calculating a denser interpolated hyperplane. Localization is then performed by estimating the projection of the momentary panoramic robot's view in the eigenspace and then by searching for the best match on the hyperplane.

A severe limitation of the standard projection method is the sensitivity to noise and occlusion in the process of recognition. This is because the momentary input image is usually used as a complete pattern when calculating the projections. If the image is occluded or noisy, the projections calculated will be far from the correct result. To overcome this, a robust algorithm for the retrieval of parameters will be designed.

Another important question when using panoramic images is how to distinguish between images that are taken under different orientations of the robot. Such images have the same pictorial content, yet rotated for an angle (phase) of ϕ. We discuss that the approaches used until now fail to simultaneously achieve insensitivity to in–plane rotation and robustness.

As the principal contribution, we propose a specific eigenspace repre-

sentation, called "*eigenspace of spinning–images*", which makes localization simultaneously robust and insensitive to in–plane orientation of the robot. The representation exploits the fact that a set of rotated templates carrying identical pictorial information can be compressed in the eigenspace in an efficient way. Further, we show that a significant compression ratio can be achieved even when integrating a number of rotated images that all represent a single location. By doing so, we also have the advantage of being able to match the incoming image directly to the whole set of rotated images in the recognition phase (Figure 1.2).

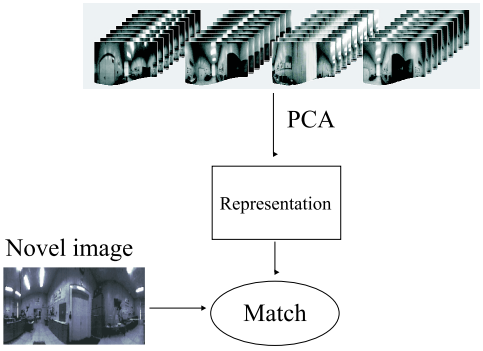


Figure 1.2: The eigenspace of spinning images allows to match the novel input image with a collection of images in multiple orientations.

The paper is organized as follows. We start with a review of the related work and, in Section 3, continue with the principles of localization using a panoramic eigenspace approach and stress its advantages and drawbacks. In Section 4 we describe how to efficiently perform the learning phase, where we construct a set of basis images which span an eigenspace that describes all the possible orientations of images in a compact and fast–to–compute fashion. In this way, we obtain the "*eigenspace of spinning–images*", which we use as the base model for the estimation of the momentary position of the robot in the localization phase. In Section 5 we show how it is possible to do that even in the presence of noise and occlusion by introducing a robust method for projecting the images on the eigenspace. Finally, in Section 6 we give the results of the experiments of localization in several indoor environments. We conclude with a summary and an outline of our ongoing work.

Chapter 2

Related work

Back in 1996, Ishiguro et *al.* [9] used 2D images for localization of a mobile visual system and, in order to compress the images, they use the subspace of Fourier harmonics, i.e., they calculate the Fourier coefficients to represent images in a lower-dimensional subspace. They refer to the appearance–based model as to "the iconic memory of scenes". They already employ a panoramic sensor and unwarp the images to a cylindrical representation (Figure 3.1). The coefficients indicate the features of global views at the reference points and the phase components indicate the orientation of the views. The experiments were made in an office environment, which was represented by a map based on an energy spring model. However, they comprise only place recognition by indexing and do not try to generate an interpolated model. The representation used is non–robust, since the Fourier transform is inherently a non–robust transformation. An occlusion in the image influences the frequency spectra in a non-predictable way and therefore changes the coefficients arbitrary.

An interesting algorithm which is based on the encoding of non-panoramic images in the parametric eigenspace was later proposed by Maeda et *al.* [17]. Recognition is based on an active vision strategy which helps to resolve the ambiguities of eigenspace analysis by deducing the position from several neighboring views.

Aihara et *al.* [2] were probably the first to use the panoramic eigenspace approach. In order to avoid the problem of rotation of the sensor around the optical axis, they used row-autocorrelated transforms of cylindrical panoramic images. They tested the system first in an indoor corridor and then by mounting the camera on a rooftop of a van and making a tour of their campus. The approach suffers from less accurate results for images acquired on

novel positions, since by autocorrelating the images some of the information is lost. Moreover, the process of autocorrelating the image is non–robust, meaning that any occlusion in the image may result in an erroneous localization.

In our first report on experiments with panoramic eigenspaces of unprocessed images [11], we suggested that alternative representations have to be found in order to handle the problem of in–plane orientation of the sensor. Independently, Pajdla and Hlaváč [22] proposed to estimate a reference orientation from images alone with the Zero Phase Representation (ZPR). ZPR, in contrast to autocorrelation, tends to preserve the original image content while at the same time achieving rotational independence, as it orients images by zeroing the phase of the first harmonic of the Fourier transform of the image. The experiments indicated that images taken at nearby positions tend to have the same reference orientation. The method is however sensitive to variations in the scene, since it operates with only one frequency using a global transform.

Both Aihara and Pajdla tried to overcome the problem of multiple orientations by applying a global transform to the image. It is, however, well known that the eigenspace methods are inherently not robust when global transformations of images are used [15]. Imagine that we obtain an image of environment in absence of people and one that is being partially occluded by persons present in the room. Either autocorrelation or Fourier transform of both images would take quite different values.

Recently, Paletta et *al.* [23] robustified the panoramic eigenspace approach by applying a Bayesian reasoning over local image appearances, which helps in rejecting false hypotheses that do not fit the structural constraints of the corresponding spline. They achieved robustness and handled rotation by dividing the panoramic image into overlapping vertical strips representing unidirectional camera views. Distributions of sector images in eigenspace are represented by mixture of Gaussians to provide a posterior distribution over potential locations. However, the local windowing introduces ambiguities that have later to be resolved through Bayesian probabilistic framework.

Gaspar et *al.* [7] used panoramic images unwarped to a birds eye view. This was useful for orienting the images in an indoor environment such as a corridor, since it could be aligned using straight lines. They claimed that a wide field of view provides enough stability for the basic PCA algorithm.

Vlassis et *al.* [28] augmented the learning phase by using a supervised technique which resulted in an optimized feature space. They did not address the problems of occlusion and rotation that are of our primary interest.

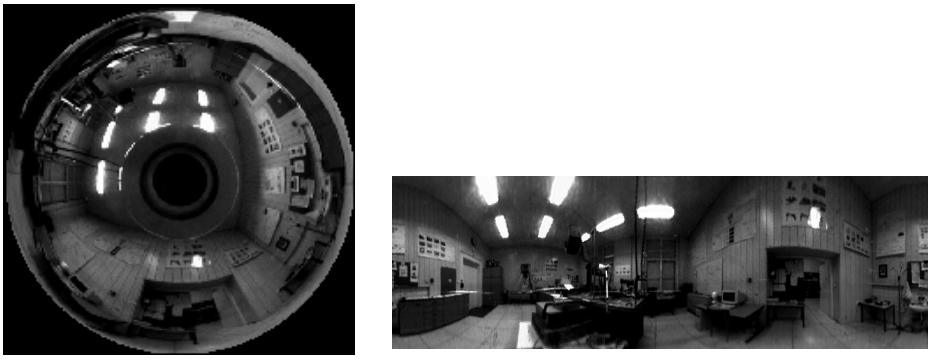
From this review of the related work we can conclude that the problems of rotation and robustness were thoroughly discussed only in the Paletta's work. While their method proved to be successful, the method that we are going to introduce solves the problem on the level of the representation, as it provides a generalized representation of rotation and applies the basic parametric eigenspace recognition schema. The main advantage of our "*eigenspace of spinning–images*" representation is that it preserves the appearance while solving the problem of rotation. The issue of robustness is addressed by a modified algorithm for the calculation of the parameters of the novel input image which is based on the selection of points that are not part of occluded regions.

Chapter 3

The principles of localization using a panoramic eigenspace

For building an appearance–based model from a set of images, the eigenspace approach proved itself as a viable one. This approach is a twofold procedure: in the *learning stage*, several panoramic snapshots are acquired which together form a good depiction of the environment. These images are taken when the robot is exploring the environment either by following some training strategy or under the supervision of a human operator. Then, an orthonormal basis system is computed, which allows us to represent every image of the training set with only a few parameters. PCA offers means for calculating an optimal basis system using Singular Value Decomposition (SVD) of the covariance matrix, representing the images in a low–dimensional subspace that is an optimal linear approximation of the original set in the least squares sense. This allows for efficient memory consumption, efficient matching and recognition. The training images have to be collected first in order to build the eigenspace. Alternatively, one can employ an incremental algorithm, where images are being sequentially integrated in the model [3].

Let us take images on a regular grid with a density that is reasonable for indoor spaces. The panoramic images are usually taken with a catadioptric camera which produces round images like the one depicted in Fig. 3.1(a). In practice, we unwarp these images to a cylindrical shape like the one depicted in Figure 3.1(b). The reason to do so is that the rotation of a round panoramic image translates to a row wise shift in a cylindrical image. It is therefore possible to virtually simulate the in–plane rotation of the camera by just shifting the rows.



(a) (b)

Figure 3.1: (a) Panoramic image taken with a spherical mirror. (b) The same image unwarped to a cylindrical image.

When all of the images are collected by the robot, we have to calculate the eigenspace which will efficiently describe the acquired data. We therefore represent images from the training set as normalized image vectors, from which the mean image is subtracted, $\mathbf{x}_i$; $i = 0 \ldots N-1$. These vectors form an image matrix $X \in \mathbb{R}^{n \times N}$

$$X = \left[\begin{array}{cccc} \mathbf{x}_0 & \mathbf{x}_1 & \ldots & \mathbf{x}_{N-1} \end{array} \right] , \tag{3.1}$$

where n is the number of pixels in the image and N is the number of training images.

The most straightforward way to calculate the eigenvectors is by solving the SVD of the covariance matrix $C \in \mathbb{R}^{n \times n}$ of these normalized vectors

$$C = XX^T = \left[\begin{array}{cccc} \mathbf{x}_0 & \mathbf{x}_1 & \ldots & \mathbf{x}_{N-1} \end{array} \right] \left[\begin{array}{c} \mathbf{x}_0^T \\ \mathbf{x}_1^T \\ \ldots \\ \mathbf{x}_{N-1}^T \end{array} \right] = VLV^T \ . \tag{3.2}$$

The columns of V are the eigenvectors $\mathbf{v}_i$, $i = 0, \ldots, N-1$, which form an orthogonal basis set. The values λ_i in the diagonal matrix L are the corresponding eigenvalues, which denote the overall variance that each of the eigenvectors encompasses.

Since the number of pixel elements n in an image is usually high, the computation of the matrix C is a time consuming task of high storage and computational demands. However, it is possible to formulate the equations

in such a way that it becomes sufficient to calculate the eigenvectors $\mathbf{v}'_i$, $i = 0, \ldots, N-1$, of the inner product matrix $Q \in \mathbb{R}^{N \times N}$

$$Q = X^T X = \begin{bmatrix} \mathbf{x}_0^T \\ \mathbf{x}_1^T \\ \ldots \\ \mathbf{x}_{N-1}^T \end{bmatrix} \begin{bmatrix} \mathbf{x}_0 & \mathbf{x}_1 & \ldots & \mathbf{x}_{N-1} \end{bmatrix} . \tag{3.3}$$

Since the eigenvectors $\mathbf{v}'_i$ are the solution of $X^T X \mathbf{v}'_i = \lambda'_i \mathbf{v}'_i$, we can calculate the eigenvectors of XX^T by $XX^T X \mathbf{v}'_i = \lambda'_i X \mathbf{v}'_i$ [4]. In this way, we derive the eigenvectors $\mathbf{v}_i$ of the covariance matrix just by projecting the $\mathbf{v}'_i$ on the set of images,

$$\mathbf{v}_i = \frac{1}{\sqrt{\lambda'_i}} X \mathbf{v}'_i \ . \tag{3.4}$$

The eigenvectors corresponding to non-zero eigenvalues of the covariance matrix form the basis of the eigenspace, which reflects the training images in the subspace of a maximum of N dimensions. If we sort the eigenvectors according to the eigenvalues, we get an ordering which reflects the principal directions of variance in the learning set of images. We can therefore choose just a subset of p eigenvectors with the largest eigenvalues to be included in the model. Each image gets therefore optimally approximated in the least squares sense up to the degree of error we make with cutting off the least informative eigenvectors. Namely, every training set image $\mathbf{x}_i$ projects into some point $\mathbf{q}_i$ in the *eigenspace*, spanned by the selected eigenvectors [20], and can be approximated by

$$\mathbf{x}_i \approx \sum_{j=1}^{p} q_{ij} \mathbf{e}_j \ . \tag{3.5}$$

The major advantage of the eigenspace method is that it allows to perform image matching in a much lower–dimensional space. Matching of images is usually performed by using some similarity or distance metric. In the image space, the city-block distance (L_1 norm) and the Euclidean distance (L_2 norm) are among the most common distance measures, while covariance and correlation are among the most common similarity measures.

In Figure 3.2 we show a graph depicting the covariance of panoramic images taken at different positions with the panoramic image that was taken at the position X=440 cm, Y=660 cm . As we see, covariance drops

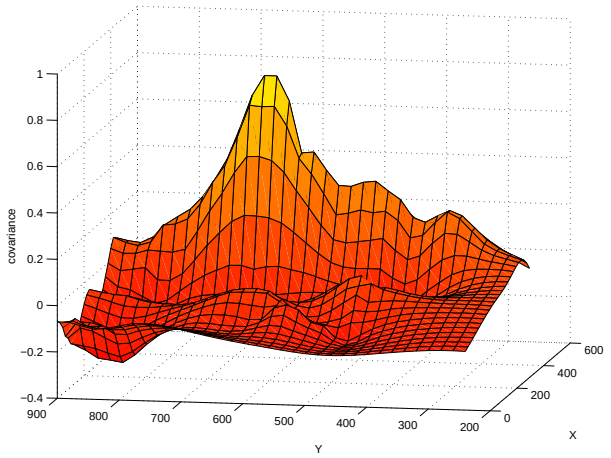


Figure 3.2: Covariance of oriented panoramic images in space. XY plane represents coordinates of the experimental environment. Measured and then interpolated is the covariance with image at X=440 cm, Y=660 cm.

with distance; position could be therefore inferred by finding the image in the training set that shows the largest covariance with the momentary panoramic view. However, it can be shown that by projecting an image in the eigenspace, the L_2 norm and the covariance produce the same result as when calculated on original images. Of course, this only holds when we use the complete set of eigenvectors. It is therefore possible to match images in a much lower–dimensional space; usually it is done by representing all the images by their projections in the eigenspace and then by adopting the L_2 norm or the Mahalanobis distance between the projections as the distance metric. Furthermore it is possible to densely interpolate the set of points in the eigenspace to obtain a hyperplane that represents an approximation of an arbitrary dense set of images [20].

To localize the robot we therefore have to find the appropriate parameters of the input image and then search for the nearest point on the hyperplane, which represents the model built on the basis of the images in the training set.

Chapter 4

Eigenspace of Spinning Images

In the previous section we have described how a panoramic eigenspace model can be used for the localization task. As we pointed out, we perform recognition as finding the nearest projection of an image on the hyperplane. The measure of distance in our case is the Euclidean distance of corresponding projections in the eigenspace, which relates to the sum of squared difference between the two images in the image space. For the recognition algorithm to be successful it is therefore crucial that all of the images taken at nearby positions are also neighbors in the eigenspace. However, for panoramic images this is true only when they are aligned in the same direction. In Figure 3.2 we already showed a graph of correlation between equally oriented images, which depicts how correlation gradually decreases with the distance. But what happens if the images are taken in an arbitrary orientation? It is intuitive that the covariance of the neighbor images would drop. We can show with a simple experiment how the distribution of the covariance changes: In Figure 4.1 we depict the covariance with the image taken at the position X=440, Y=660 for the set where images were randomly oriented. One can see, that in this case covariance is not a reliable indicator of the distance between the locations.

Therefore, if we have access to an internal compass that indicates the orientation of the sensor, we can use its information in order to adequately orient the images. However it is possible to overcome this problem by only using the visual input.

Several of the approaches that we reviewed in Section 2 dealt with the problem of in–plane rotations of panoramic images [2, 22, 9]. Driven by a

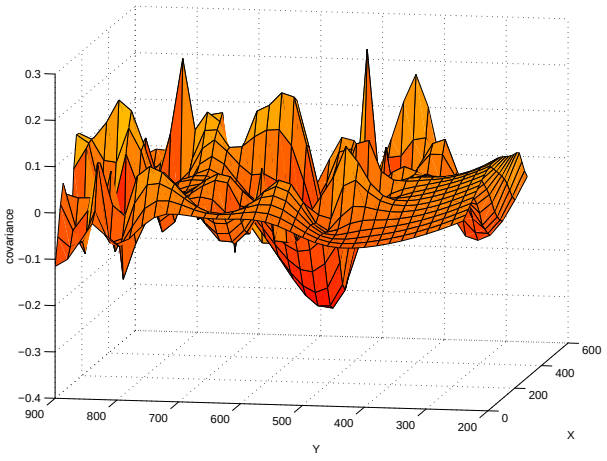


Figure 4.1: Covariance of randomly oriented panoramic images in space. XY plane represents coordinates of the experimental environment. Measured and then interpolated is the covariance with image at X=440, Y=660.

desire to obtain a compact representation, the general tendency of the approaches applied until now has been to represent each position with just one image. Thus, different representations have been proposed that achieve some sort of rotational-invariance, which results in a shift-invariance for cylindrical panoramic images. One way to achieve the invariance is by applying a transformation that produces the same output regardless of the shift in the input image. Such a transformation is autocorrelation (which in the particular case of cylindrical panoramic images can also be applied row-wise [2]). Another approach is to orient the images in a reference orientation, and thus preserve appearance [22].

We argue that while these approaches achieve shift-invariance, they can not be used for robust localization. This is due to the fact that they use a global transform of the original images that is sensitive to changes in the scene.

What we propose in this paper is a representation that includes all the variations in rotation that we can simulate by shifting the cylindrical panoramic images row wise. Multiple orientations of the robot at the training positions are then represented as points in the eigenspace. As we show, the moderate increase in storage complexity pays off with robustness of position retrieval. Furthermore, the representation that we obtain by including all the rotated images in the training set is partially analytical; points cor-

responding to one position but under different orientations lie on analytical hyperplanes, which makes localization easier. In this way we achieve rotation invariance without the need for searching for the correspondences or applying a global transform.

4.1 Eigenspace representation of a set of rotated versions of one panoramic image

Let us first assume that all of the images in the training set capture the same scene from *a single point of view*, but under different in–plane rotation. If we unwarp the panoramic image to a cylindrical projection, the rotation can be expressed simply as a rowwise shift (Figure 3.1). Since the images are uniformly rotated (shifted), each image can be generated by rotating (shifting) the original image $\mathbf{x}_0$ for $2\pi/N$.

Uenohara and Kanade [27] showed that in the case of the image set consisting of rotated examples of one original image, the inner product matrix Q is a symmetric Toeplitz matrix. Since Q is also circulant, its eigenvectors $\mathbf{v}'_i$ are not dependent on the contents of the images [27]. Q is of the form

$$Q = \begin{bmatrix} q_0 & q_1 & \cdots & q_{N-2} & q_{N-1} \\ q_{N-1} & q_0 & q_1 & \cdots & q_{N-2} \\ q_{N-2} & q_{N-1} & q_0 & q_1 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ q_1 & \cdots & q_{N-2} & q_{N-1} & q_0 \end{bmatrix} . \tag{4.1}$$

It can be derived from the *shift theorem* [10], that the eigenvectors of a general circulant matrix are the N basis vectors from the Fourier matrix $F = [\mathbf{v}'_0, \mathbf{v}'_1, \ldots, \mathbf{v}'_{N-1}]$

$$\mathbf{v}'_k = \left[1, w^k, w^{2k}, \ldots w^{(N-1)k}\right]^T, \quad k = 0, \ldots, N-1; \tag{4.2}$$

where $w = e^{-2\pi j/N}$, $j = \sqrt{-1}$.

The eigenvalues can be calculated simply by retrieving the magnitude of the DFT of one row of Q;

$$\lambda'_k = \sum_{i=0}^{N-1} q_i e^{-2\pi j \frac{ik}{N}} . \tag{4.3}$$

This interesting property also emphasizes the central point of the Fourier analysis, as it indicates that the Fourier basis diagonalizes every periodic constant coefficient operator, in our case the circular shift operator [26].

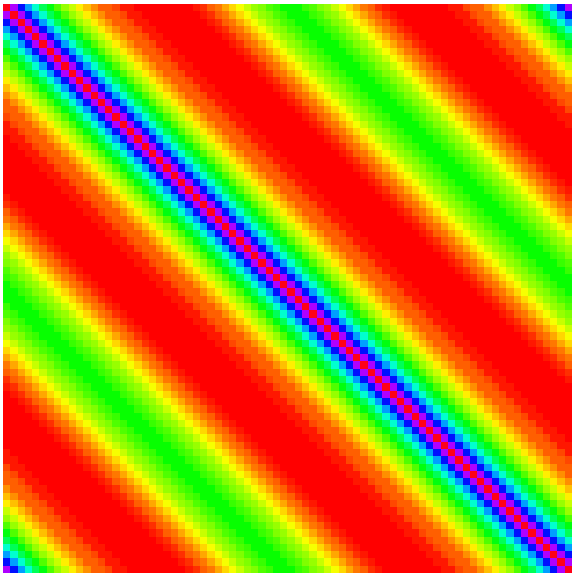


Figure 4.2: The inner product matrix Q for rotated images captured at a single location.

Since $q_i = q_{N-i}$, our matrix is circulant symmetric, and therefore we can choose an appropriate set of real–valued orthogonal eigenvectors. As it turns out, the proper basis are the cosine functions from the real and the sine functions from the imaginary part of the Fourier matrix [26].

We can therefore compute the eigensystem of Q just by first computing the autocorrelation vector $[q_1, q_2, \ldots, q_{N-1}]$, and then by calculating the λ'_i values, which should be afterwards sorted by decreasing magnitude. The eigenvectors $\mathbf{v}'_i$ corresponding to k largest eigenvalues can then be easily selected from the corresponding basis vectors of the Discrete cosine transform (DCT) [27]:

$$v'_{km} = \cos\left[\frac{\pi(2m+1)k}{2N}\right]; \quad \begin{array}{rcl} m & = & 0, \ldots, N-1 \\ k & = & 0, \ldots, N-1 \end{array} \tag{4.4}$$

Thus, with the help of the DCT, it is possible to compute the basis vectors much more efficiently.

4.2 An extension to a set of rotated panoramic images acquired at different locations

When dealing with the problem of localization of a mobile robot [12], we need to encode P images from P locations, each of them being rotated N times (Figure 1.2). In this case, we cannot directly apply the previous approach to the calculation of eigenvectors of circulant matrices, since the inner product matrix A,

$$A = X^T X = \begin{bmatrix} Q_{11} & Q_{12} & \dots & Q_{1P} \\ Q_{21} & Q_{22} & \dots & Q_{2P} \\ \dots & \dots & \dots & \dots \\ Q_{P1} & Q_{P2} & \dots & Q_{PP} \end{bmatrix}, \tag{4.5}$$

is composed of several circulant blocks Q_{ij}, which are, in general, not symmetric. An illustration of A for 5 locations is shown in Figure 4.3. However, as we will show, it is still possible to calculate the eigenvectors without performing the SVD decomposition of A.

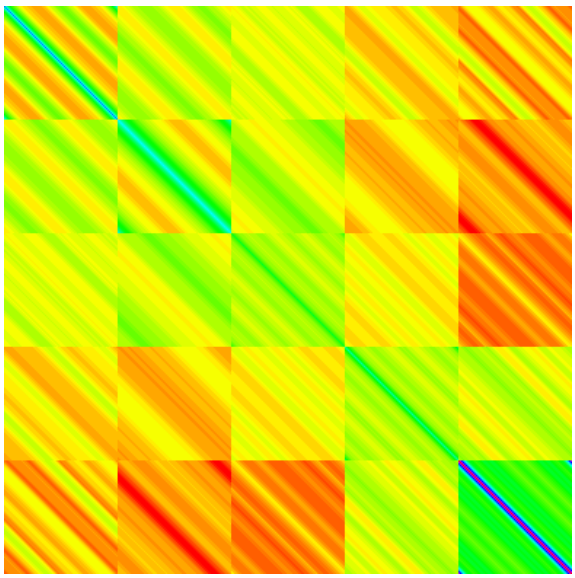


Figure 4.3: The inner product matrix A for 5 locations.

We have to solve the eigenvalue problem

$$A\mathbf{w}' = \mu\mathbf{w}', \tag{4.6}$$

where $(\mu, \mathbf{w}')$ is the eigenpair of A. The fact that the matrix blocks Q_{ij} of A are circulant matrices is crucial. As it was already mentioned, every circulant matrix can be diagonalized in the same basis by Fourier matrix F. Consequently all the matrices Q_{ij} have the same set of eigenvectors $\mathbf{v}'_k = \left[1, w^k, w^{2k}, \ldots w^{(N-1)k}\right]^T$, $k = 0, \ldots, N-1;$. We shall find the eigenvectors $\mathbf{w}'$ of A among the vectors of the form

$$\mathbf{w}'_k = \left[\alpha_{k1}\mathbf{v}'^T_k, \alpha_{k2}\mathbf{v}'^T_k, \ldots \alpha_{kP}\mathbf{v}'^T_k\right]^T \quad k = 1 \quad \ldots \quad N \quad . \tag{4.7}$$

Equation (4.6) can be rewritten blockwise as

$$\sum_{j=1}^{P} Q_{ij}(\alpha_{kj}\mathbf{v}'_k) = \mu\alpha_{ki}\mathbf{v}'_k, \quad i = 1 \quad \ldots \quad P \quad . \tag{4.8}$$

Since $\mathbf{v}'_k$ is an eigenvector of every Q_{ij} the equations simplify to

$$\sum_{j=1}^{P} \alpha_{kj}\lambda'^k_{ij}\mathbf{v}'_k = \mu\alpha_{ki}\mathbf{v}'_k, \quad i = 1 \quad \ldots \quad P, \tag{4.9}$$

where λ'^k_{ij} is an eigenvalue of Q_{ij} corresponding to $\mathbf{v}'_k$. This implies a new eigenvalue problem

$$\Lambda\boldsymbol{\alpha}_k = \mu\boldsymbol{\alpha}_k \quad , \tag{4.10}$$

where

$$\Lambda = \begin{bmatrix} \lambda'^k_{11} & \lambda'^k_{12} & \ldots & \lambda'^k_{1P} \\ \lambda'^k_{21} & \lambda'^k_{22} & \ldots & \lambda'^k_{2P} \\ \ldots & \ldots & \ldots & \ldots \\ \lambda'^k_{P1} & \lambda'^k_{P2} & \ldots & \lambda'^k_{PP} \end{bmatrix} \tag{4.11}$$

and

$$\boldsymbol{\alpha}_k = [\alpha_{k1}, \alpha_{k2}, \ldots \alpha_{kP}]^T \quad . \tag{4.12}$$

Since A is block-symmetric, Λ is symmetric and we have P linearly independent eigenvectors $\boldsymbol{\alpha}_k$ which provide P linearly independent eigenvectors $\mathbf{w}'_k$ in (4.7). Since the same procedure can be performed for every $\mathbf{v}'_k$, we can obtain $N \cdot P$ linearly independent eigenvectors of A.

It is therefore possible to solve the eigenproblem using N decompositions of order P. Since P is usually small in comparison to the total number of

images $P \cdot N$, this method offers a similar improvement as the method in [27].

However, by looking at the properties of the circulant matrices one can deduce, that this method works only if we use the complex Fourier basis as the eigenvector set for the circulant matrix. In fact, this set of basis vectors is the only common eigenspace for all the submatrices Q_{ij} from A.

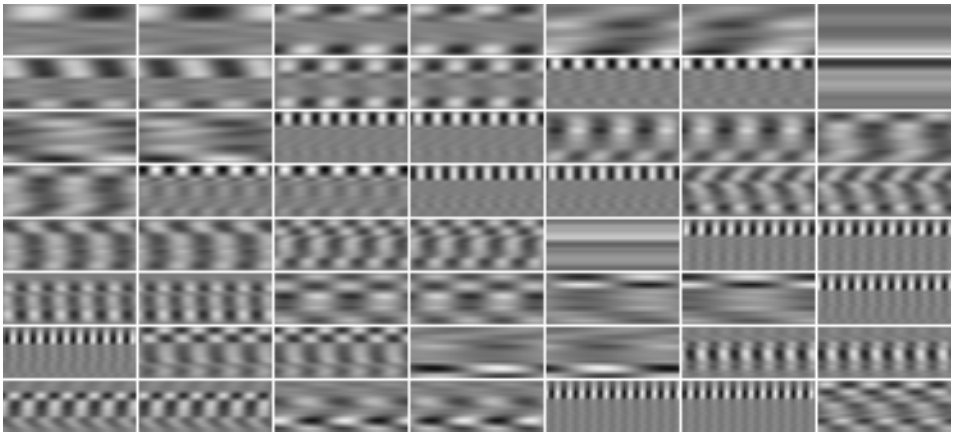


Figure 4.4: First 56 eigenvectors calculated for a set of rotated images representing 63 locations.

Those complex eigenvectors $\mathbf{w}'_k$, that are not of zero frequency (gain), come in conjugated pairs. We can therefore reduce their number by half by taking only one eigenvector of a pair. In Figure 4.4 are depicted the real and the imaginary parts of the the final eigenvectors $\mathbf{w}_i$. It can be seen that the eigenvectors have a structure resembling row wise harmonic functions (pairs have the same frequency), however every row may differ from the other in the amplitude and the phase of the signal. Indeed, these eigenimages carry the most of the pictorial information, since this set of basis functions is still the optimal linear approximation of the learning set.

To represent each training image in the eigenspace, we have to estimate its coefficients by projecting the image on the basis vectors. Since our eigenvectors are complex, the coefficient vectors are also complex, and can be viewed as points in the complex eigen–subspace. In Figure 4.5 we can see the coefficients for an image and its rotated siblings. The rotation (shift) of the image results in the change of angle of the coefficient vector in the complex eigenspace, while the magnitudes of the coefficients do not change.

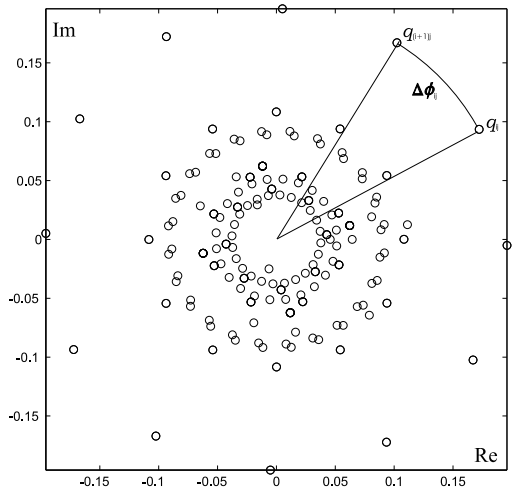


Figure 4.5: Complex coefficients corresponding to the first ten eigenvectors for one location of the training set. Depicted is the angle between the coefficients of two consecutive rotated images for the first eigenvector.

Because of this property, it is not necessary to store the coefficients for all the images. In fact, if we have just one representative coefficient for one viewpoint, all the others can be generated on the fly with a simple rotation in the complex plane.

Let $\mathbf{x}_i$ be the i-th image and $\mathbf{x}_r;\ r = i+1, \ldots, (i+N-1)$ a set of its rotated siblings. Images are evenly rotated, each by $2\pi/N$ radians regarding to its predecessor. The coefficient q_{ij} of the image $\mathbf{x}_i$ can be calculated as $< \mathbf{x}_i, \mathbf{e}_j >$. Let the j–th coefficient of the image $i+1$ be $q_{(i+1)j}$. The angle between the two coefficients is

$$\Delta\phi_{ij} = \arctan \frac{Re(q_{(i+1)j} - q_{ij})}{Im(q_{(i+1)j} - q_{ij})} \quad .$$

Every other coefficient q_{rj} can now be calculated as

$$q_{rj} = q_{ij} e^{2\pi\sqrt{-1}(r-i)\Delta\phi_{ij}} \qquad \begin{array}{rcl} r & = & i+1, \ \ldots \ (i+N-1) \\ j & = & 1, \ \ldots \ K \end{array} \quad .$$

It is therefore easy to handle rotation using a set of complex eigenvectors, since the coefficients, themselves also being complex, can be intuitively handled in the recognition stage. However, if one wishes to obtain a real basis

(this is possible since A is symmetric), this can be done in a straightforward procedure by using normalized vectors taken from the real and imaginary parts of complex eigenvectors. In this way, also the image coefficients are real, their values being the same as the real and imaginary parts of the complex counterparts. In Figure 4.6 we depict a three dimensional subspace, where the first three coefficients are used to illustrate the projections of images representing three locations. As one can see, coefficients for a single location, representing one panoramic image in several orientations, form a regular harmonic closed loop.

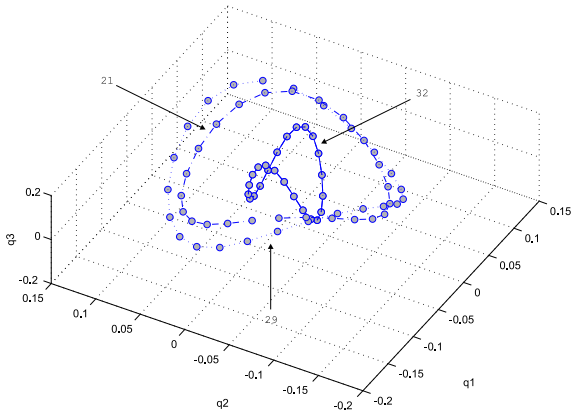


Figure 4.6: Coefficients corresponding to the first three eigenvectors for locations 21, 29 and 31 of the training set.

Chapter 5

Making localization robust

Let us now for simplicity of notation assume that the coefficients are real numbers. The standard method to recover the parameters is to project the image vector onto the eigenspace [20]:

$$q_j(\mathbf{y}) =< \mathbf{y}, \mathbf{e}_j >; \quad j = 1 \dots p \tag{5.1}$$

However, this way of calculation of parameters is non-robust and thus not accurate in the case when the input image locally deviates from the image approximated in the environment map. These deviations can sometimes be a side effect of the design of the panoramic sensor (e.g., self–occlusion of the camera holder). Also they may be caused by occlusions due to objects moving in the environment. Since panoramic images capture a wide field-of-view it is almost impossible to avoid any disturbances during the localization (operation) phase, thus it is essential that a representation of the environment map enables efficient robust matching.[1] E.g. if we imagine a mobile robot roaming around with a model acquired under a set of stable conditions, every change in environment, such as displaced objects or people walking around can result in severe variations with respect to the original images learned at the training stage.

To overcome the erroneous calculation of image parameters when the visual content deviates from the learning examples, we propose to use the robust approach [15], that, instead of using the image vectors as a whole (Equation 3.5), generates and evaluates a set of hypotheses $\mathbf{r}$ as subsets of image points $\mathbf{r} = (\mathbf{r_1}, \mathbf{r_2}, \dots, \mathbf{r_k})$. In fact, the coefficients can be retrieved

[1] Robustness is defined as the extent of the ability of a method to give expected results despite the deviations of the input data.

by solving a set of linear equations on $k = n$ points:

$$x_{r_i} = \sum_{j=1}^{n} q_j(\mathbf{x}) e_{jr_i} \quad 1 \leq i \leq n \ . \tag{5.2}$$

The principle of such computation is clearly illustrated in Figure 5.1.

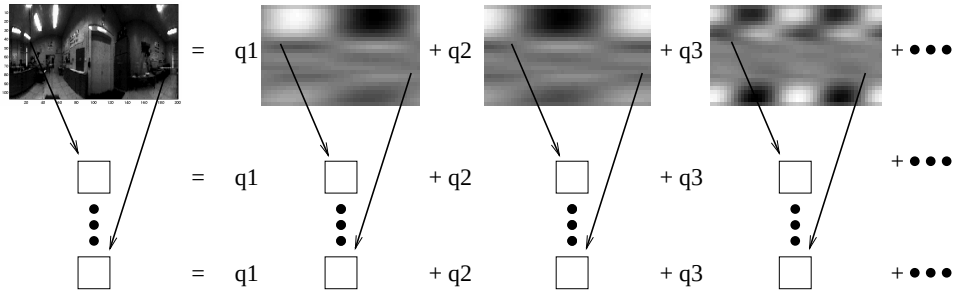


Figure 5.1: Calculating the coefficients from a set of linear equations.

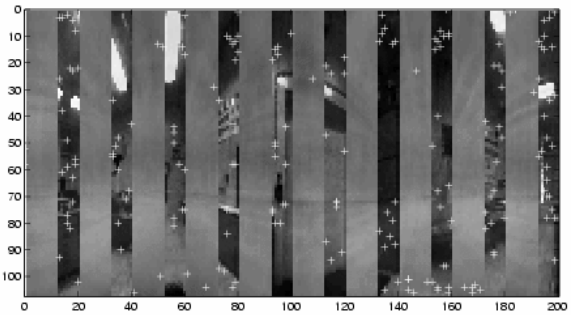


Figure 5.2: Crosses denote points that contribute to the generation of a hypothesis after three steps of α–trimming. Image is 60% occluded with vertical bars.

By selecting only p, $p \leq n$ eigenimages as our basis we have to solve an over-constrained system in a robust way, so that the solution set of parameters minimizes

$$E(\mathbf{r}) = \sum_{i=1}^{k} (x_{r_i} - \sum_{j=1}^{p} q_j(\mathbf{x}) e_{jr_i})^2 \ . \tag{5.3}$$

We solve the system on $k, k > p$ points, where k is significantly smaller than the total number of image points. The initial set of points is randomly

initiated at first. After the robust solving of the set of equations, we first perform an α–trimming step, in order to allow only the points on which the error is arbitrary small to contribute to the further computation of the parameters. The whole procedure then iterates either until the error gets small enough or until the number of remaining points meet some boundary condition. As we can see in Figure 5.2, at this stage most of the points in the occluded regions are excluded from the computation.

To increase the probability of avoiding points that are noise or represent occlusion, several different subsets of points are generated, resulting in multiple hypotheses. A hypothesis consists of a set of parameters, an error vector $\boldsymbol{\epsilon}$ calculated as the squared difference between the original image and its reconstruction $\sum_{j=1}^{p} q_{ij}\mathbf{e}_j$, and the domain of compatible points that satisfy an error margin constraint. These hypotheses are then subject to a selection procedure, based on the *Minimal Description Length* principle, as described in [15].

It is important to note at this point that the method is based on two assumptions. First, we assume, that images are not preprocessed in a way that would not allow to correctly evaluate the outliers. As an example we can mention Aihara [2] who used row-autocorrelated images, where error from outliers spreads globally over the transformed image. Secondly, the evaluation of the hypotheses is based on the reconstruction of the image, which gets evaluated against the original. Obviously, this also prohibits the use of global transforms.

Chapter 6

Experimental results

In Figure 6.1 we can see an example of an office environment with marked positions where learning images were taken during the training phase (denoted with squares). We used two different mirrors in the catadioptric camera. The first mirror was of a spherical shape, while the second mirror had a hyperbolical reflective surface of a r=1.9 ratio.[1] The round images depicted in Figure 3.1(a) were acquired with the spherical mirror setup, while the images in Figure 6.10 were acquired using the hyperbolic mirror. During our experiments we did not notice any significant difference in the performance of the two mirror shapes.

In Table 6.1 we compared the times required for building the eigenspace by 1) using the standard decomposition of the correlation matrix XX^T, 2) by calculating the decomposition of the inner product matrix X^TX and 3) by using our approach. The tests were made for images of dimensions 40×68, the latter being the width of the image. Each image was therefore rotated 68 times, i.e., for 40 locations we got 2720 images. Since this is also the number of image elements, this is the border case when the covariance matrix is of the same size as the inner product matrix, and the complexity of the SVD method reaches its upper bound.

Once the eigenspace is built, we can interpolate the memorized coefficients in order to form a hyperplane that generalizes our knowledge in order to represent the intermediate locations. An example of an interpolated hyperplane for the real coefficients corresponding to the first eigenvector is shown in Figure 6.2 (coefficients are shown for one rotation only).

In the case of complex coefficients, if we memorized only one coefficient vector per location, we can search among the scores of N projected coeffi-

[1] For a review of catadioptric cameras see [8].

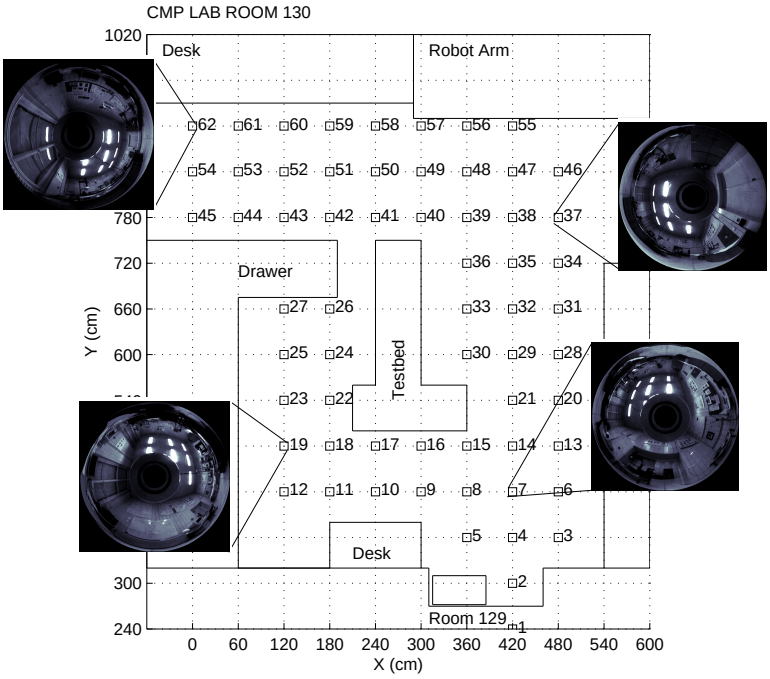


Figure 6.1: Map of $P = 62$ locations, where images were taken for training the robot. Images shown correspond to locations 7, 19, 37 and 62, respectively.

cients, i.e., we first search for the nearest neighbor of the coefficient vector corresponding to the input image and then we rotate this coefficient vector by Φ, where Φ is a vector of estimated $\Delta\phi_k$ values.

6.1 Using shift invariant properties of complex coefficients

The time complexity of the search can be quite high if the parameter space is large. It is however easy to decrease it dramatically by using a two-step search.

When using a complex eigenspace, a closer look at the coefficients reveals that the modulus of the coefficient vectors does not change for the coefficient of the original image and its rotated siblings. If $\mathbf{q}$ is the coefficient vector of

locations (P)	XX^T	X^TX	CPLX
10	2507.3	55.8	16.1
20	2569.6	429.2	105.3
30	2634.8	1400.3	312.4
40	3007.7	3252.3	853.2

Table 6.1: Timings for building of eigenspace (in seconds).

an input image, we first calculate its modulus vector $\mathbf{m} = [m_1, m_2, \ldots, m_k]$

$$m_i = \sqrt{Re^2(q_i) + Im^2(q_i)}; \;\; i = 1 \ldots k. \tag{6.1}$$

In the first search we make a list of coefficients on the hyperplane which have a modulus vector close to $\mathbf{m}$. These points and their parameters are then the only candidates for the second search. As our experiments show, this strategy dramatically speeds up the algorithm.

6.2 Results of localization

In this section we will first present experiments on localization of a mobile robot by using a standard approach with images oriented in a reference orientation. The next experiment, where an eigenspace of spinning images will be used, will demonstrate the effectiveness of our approach. We will continue with experiments that demonstrate robustness in the presence of occlusions. Finally, we will demonstrate the behavior of the method in different types of environments.

First we present the results of localization of a mobile robot equipped with a panoramic camera in an environment of roughly 6x9 meters. The training images were taken at 62 locations, denoted by squares in Figure 6.1, and 35 of them can be seen in Figure 6.3. For the first experiment an eigenspace was built by taking only one image for every position. In order to align the images as if the orientation of the robot was known, all of the images were virtually oriented in the same direction. Localization tests were then performed by taking an eigenspace of 10 dimensions onto which a set of 100 equally oriented test images was projected (Figure 6.4). The position of the robot was then deduced from the calculated parameters for each image in the test set. In Figure 6.5(a), the testing positions are denoted by filled circles, while the recovered positions are denoted by blank circles. The average error of localization was 12 centimeters.

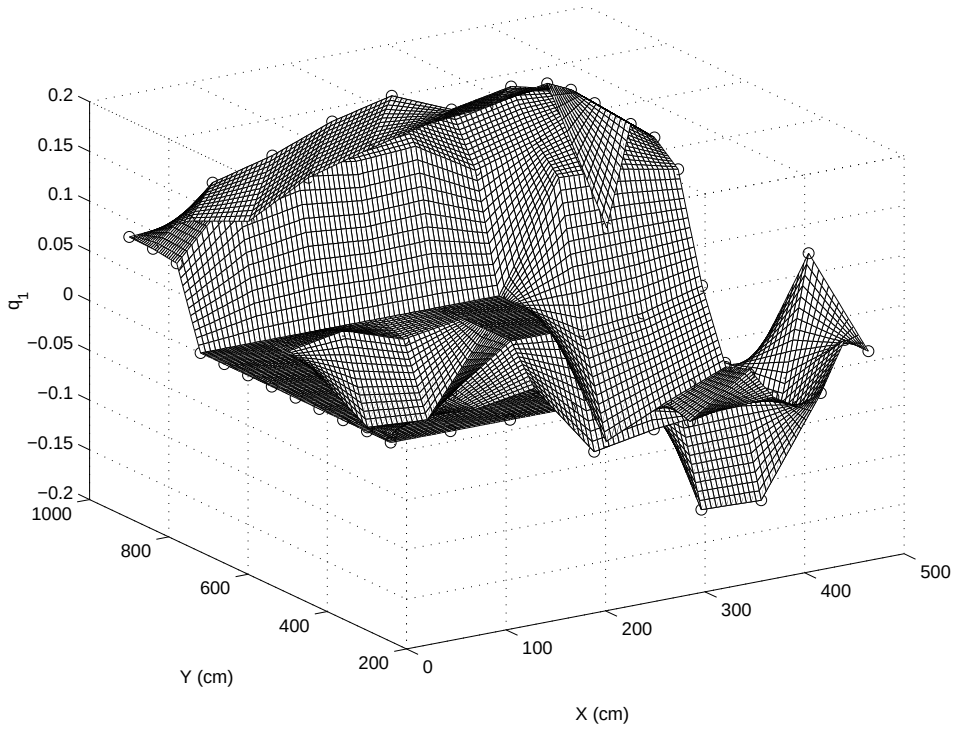


Figure 6.2: Interpolated hyperplane for the coefficients of the first eigenvector. Coefficients belonging to the images from the training set are depicted as empty circles.

The next experiment was performed in the identical environment by using an eigenspace of spinning images. Each of the 62 images in the training set was rotated 50 times, so that the interval of rotation was 7.2°. Tests were performed on a complex eigenspace of 10 dimensions and for the same testing set of 100 images (Figure 6.4). To thoroughly test the rotational invariance, the images in the testing set were now rotated randomly in order to simulate multiple orientations of the sensor. The map with the results can be seen in Figure 6.5(b). Due to an increase of the training data, the dimension of the eigenspace needed to obtain comparable results increases (the effective dimension of the eigenspace with 10 complex eigenvectors is roughly 20, since every complex vector, except those whose imaginary component equals zero, has two components), however, as we can see from the map, by doubling the dimension we get a comparable average localization error of 16 centimeters.

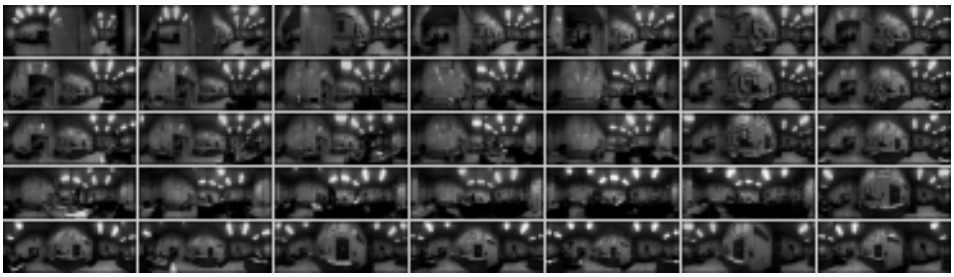


Figure 6.3: Images from the training set.

Figure 6.4: Images from the testing set.

6.3 Experimental Results for the Robust Method

Since in the original set of images there is no significant occlusion (besides some change in the local illumination of the windows area), the standard method has a small error, mostly coming from the inaccuracy of the model itself. In fact, as it can be seen from the graphs in Figure 6.5, the mean error of localization is between 11 cm and 16 cm for 0% occlusion.

The performance of the robust method at higher levels of occlusion noise is compared to that of the standard method in Figure 6.6. We can see a significant improvement in precision as a result of applying the robust method. Even in situations when more than half of the surrounding is occluded, the robust method retrieves positions that are reasonably close to the correct ones. This can be clearly seen in Figure 6.7. On the left we can see that the standard method breaks down because of the occlusion in the testing data, while the results of the robust estimator on the right show quite regular localization results with mean error under 60 cm.

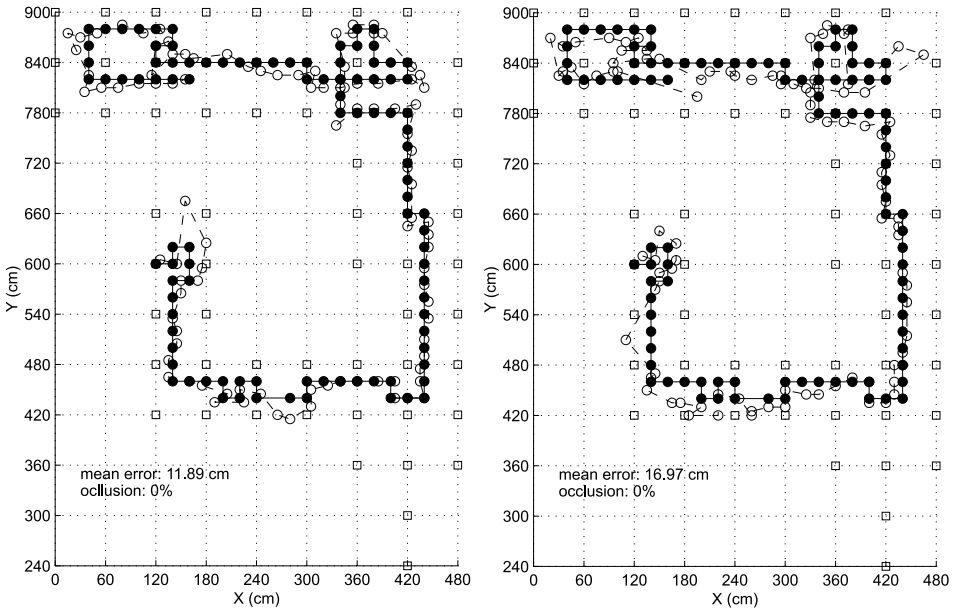


Figure 6.5: Results of localization for (a) the standard eigenspace approach using a 10 dimensional eigenspace; and (b) the eigenspace of spinning images using 10 complex eigenvectors. Black dots denote the test locations, while empty dots denote the estimated locations of the robot.

6.4 Results in different types of environments

To further test the algorithm we performed another set of experiments in a smaller area. The robot was roaming in a space of 2x3 meters and taking images at predefined time intervals. That resulted in a set of 162 images that were then divided into two sets. The first set containing 108 images was used for training, while the remaining 54 images were used for testing the localization. In Figure 6.8(a) we can see the measured locations of the testing images. The localization was first performed using a non robust estimation of image parameters, which resulted in a set of estimated positions depicted on Figure 6.8(b). The only disturbance in this set of images is the holder of the mirror, which represents a minor self–occlusion. Nonetheless, the results get considerably improved by using the robust method (Figure 6.8(c)). In a smaller space the relative value of the error increases, however it is still around 10 centimeters on the average. This performance is still satisfactory, as these are just the direct estimates, without using any of the knowledge

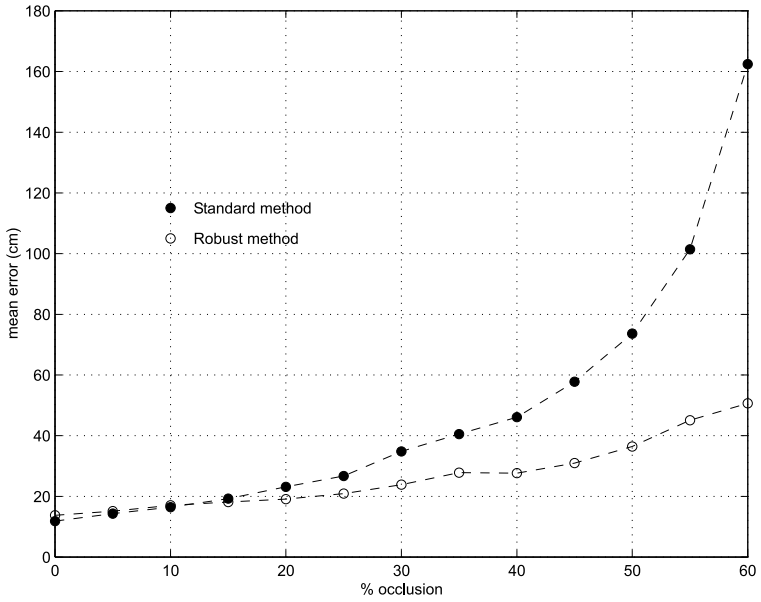


Figure 6.6: Mean error of localization for the standard and for the robust method.

on neighboring measurements to improve accuracy.

Next we questioned, how does the method perform when training images are taken at close intermediate intervals, so that the variance of the visual information captured in the training set is relatively small since panoramic images get very similar in appearance. In order to test this, we built a spinning images eigenspace by sampling locations that were no more than 15 cm apart. Then we sampled 40 test images, lying close together in a radius of approximately 1 meter; these locations are denoted by empty circles in Figure 6.9. The final recovered positions are denoted in the same figure as dots. The graph shows that even at a smaller scale the localization is stable and reliable.

Another experiment was made in a large entrance hall, which measures several dozens of meters in each direction. We acquired images on a path which measured approximately six meters in length. Afterwards, we selected every fourth image as a training example. Finally, we run the localization algorithm on all the 161 images using a 10 dimensional subspace. The average error in this case was only 4 cm. Both the original and the recovered path can be seen in Figure 6.11.

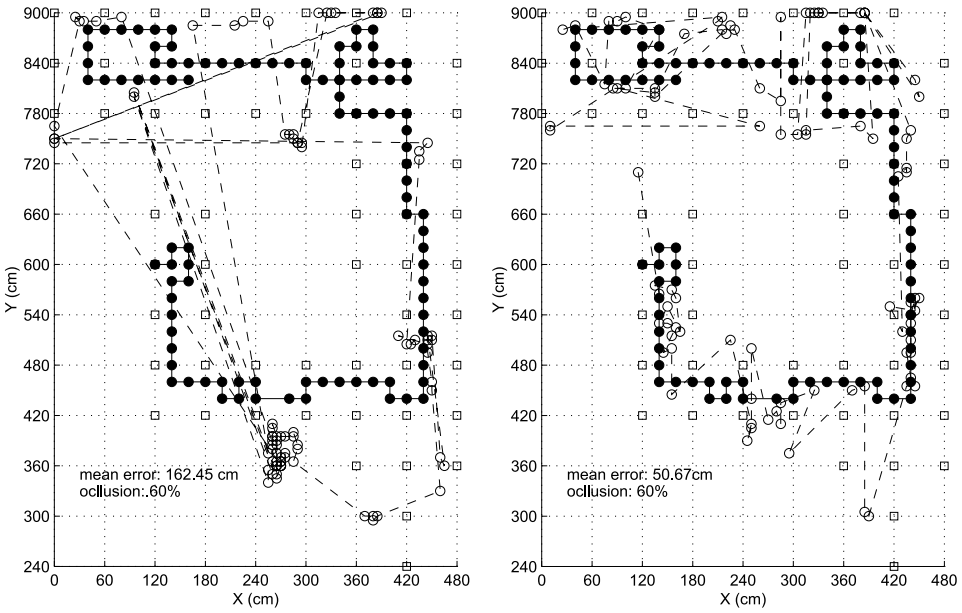
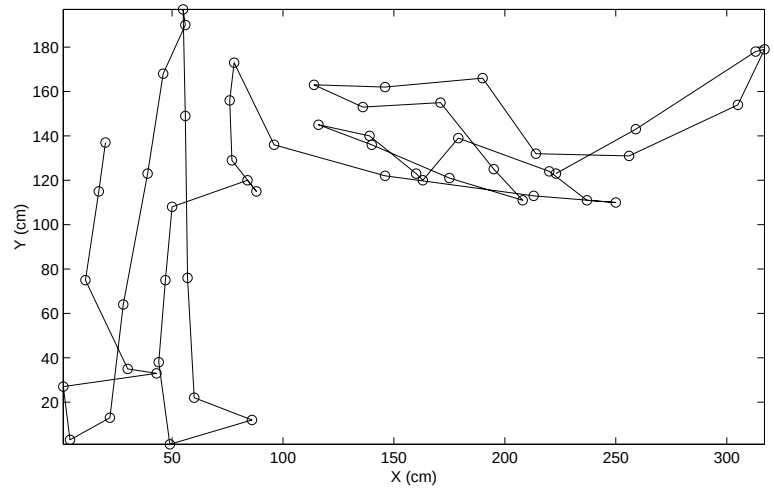
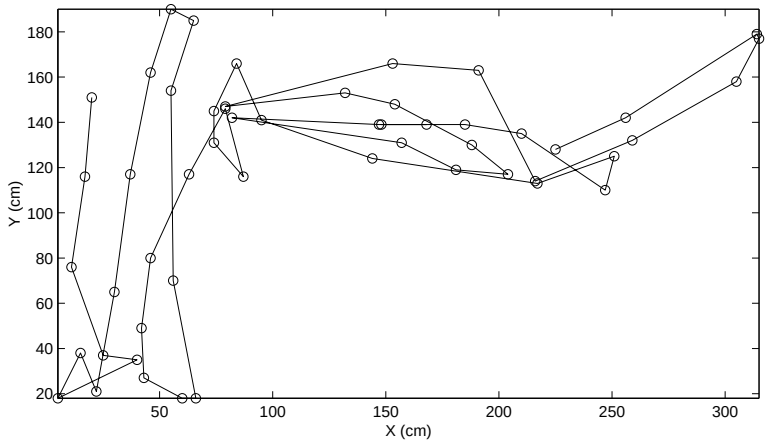


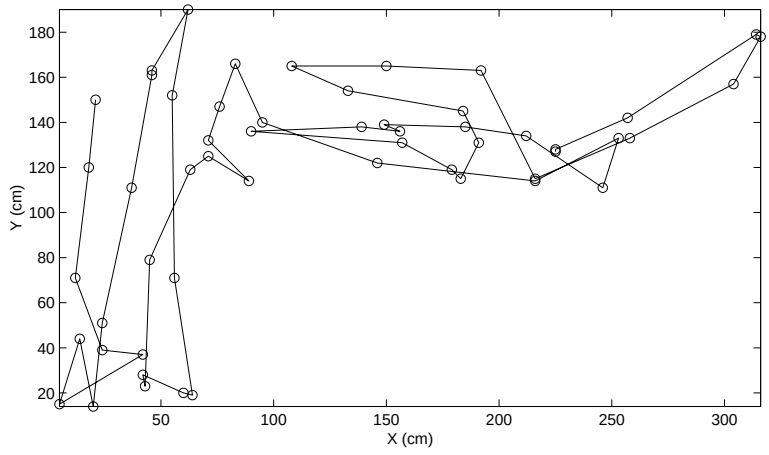
Figure 6.7: Localization on an imaginary path of 100 images at 60% occlusion. Left: standard method; right: robust method.



(a)



(b)



(c)

Figure 6.8: Localization on an imaginary path of 54 images. (a) true positions, (b) positions estimated by the standard projection; (c) positions estimated with the robust algorithm.

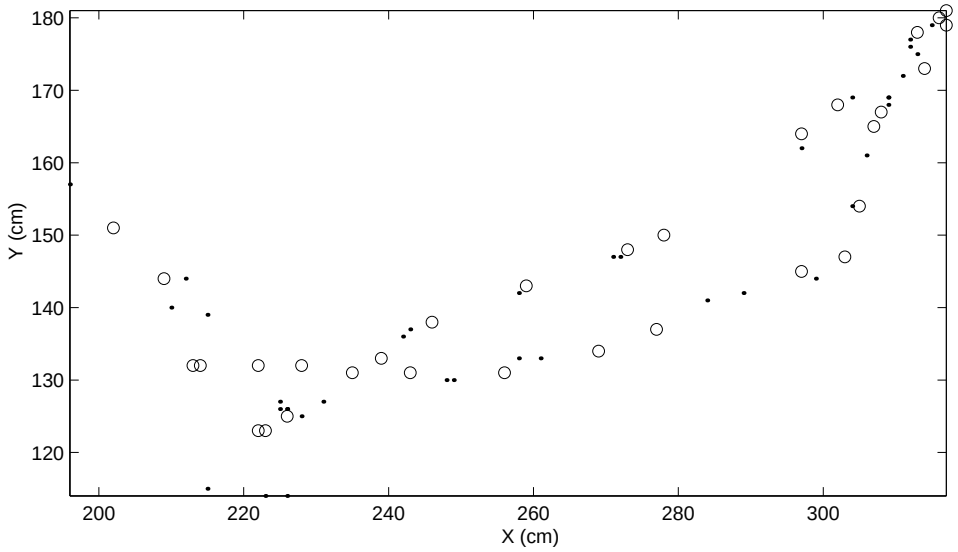


Figure 6.9: Localization on a dense set of 40 images with the robust algorithm.

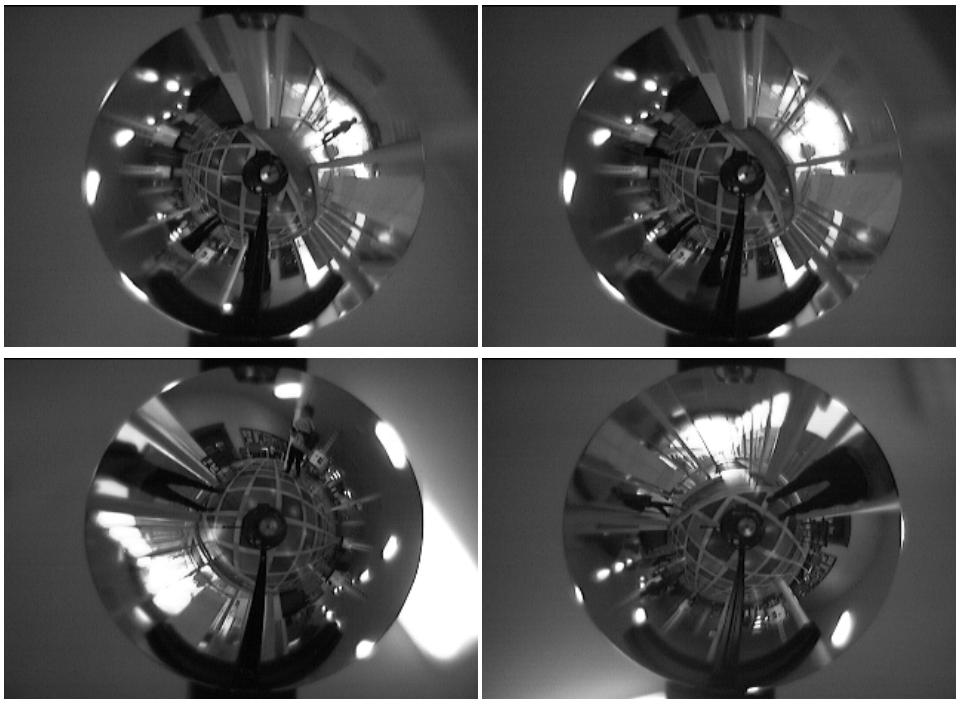


Figure 6.10: Five example of the robot's view from the path where 161 images were taken. Some of the images in the testing set contain people walking in front of the camera.

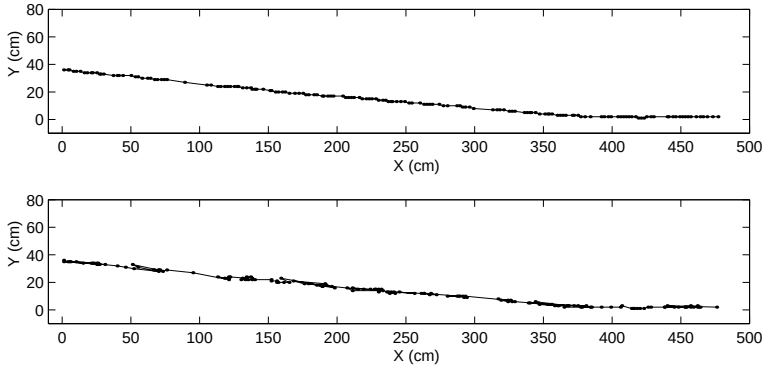


Figure 6.11: Localization on a path of 161 images, where 41 images were used for training. (a) original path; (b) recovered path.

Chapter 7

Conclusion

In this paper we presented a framework for visual self–localization of mobile robots using an appearance–based parametric model build from panoramic snapshots of the environment. We proposed the "*eigenspace of spinning–images*" as a viable representation that solves many of the problems of the standard appearance–based techniques. It efficiently encodes the varying orientations of the sensor and the resulting eigenvectors provide an analytical framework for the computation of parameters where rotation is being handled easily. This allows for robust localization under arbitrary orientations of the robot in the case when no external source of information on orientation is available to align the images. We showed how to efficiently calculate the optimal subspace of a set of training snapshots even without actually decomposing the covariance matrix. We also introduced a robust method for the retrieval of parameters which enables robust localization even in the presence of noise and occlusion. In this way, the robot is capable of localizing itself in dynamic and unstructured environments.

Our current ongoing work is concentrated mainly on the incremental learning, which enables us to bridge the gap between the learning and the training stage [3]. This coincides with many of the efforts in the robotics community intended to provide frameworks that allow for simultaneous localization and modelling (SLAM).

Among the many advantages that an incremental algorithm brings we concentrate on the possibility of the dynamical building of multiple eigenspaces, where each of the eigenspaces would represent a locally coherent apart of the environment [16]. Another problem of the classical eigenspace approach that we are trying to solve is the non–robustness of the learning phase. We are currently in the phase of developing algorithms for the robust building

of the eigenspaces.

Lately, we achieved significant progress regarding the problem of changing illumination. As we showed, such a variation in appearance can not be solved satisfactorily solely by the robust method of estimation of parameters. We therefore successfully applied a novel method that is based on filtering the eigenspace with a set of gradient filters [5].

The primary intent of the results presented in this paper is to show the capability of the appearance–based localization. They all show results obtained solely by the calculation of the parameters of the momentary input image. In a real–world implementation, this system would be preferably integrated in a probabilistic framework which would account also for temporal coherence of the results, by using additional knowledge on the local robot movements. We therefore intend also to put efforts in the development of such a system as we believe that this will influence the stability and the precision of localization in a positive way.

Zahvala

Zahvaljujem se vsem, ki so pripomogli k izvedbi naloge, posebej mentorju doc. dr. Alešu Leonardisu za pomoč, motivacijo in koristne informacije, Tomášu Pajdli in Janu Černíku iz laboratorija CMP v Pragi za testne panoramske slike, članom Laboratorija za računalniški vid pa za praktične nasvete in diskusije. Prav posebna zahvala gre tudi vsem, ki so me spremljali in podpirali v času študija, zlasti še staršem.

Bibliography

[1] *IEEE Computer Society Workshop on Models versus Exemplars in Computer Vision,* Kauai, Hawaii. IEEE Computer Society, December 2001.

[2] H. Aihara, N. Iwasa, N. Yokoya, and H. Takemura. Memory-based self-localisation using omnidirectional images. In Anil K. Jain, Svetha Venkatesh, and Brian C. Lovell, editors, *14th International Conference on Pattern Recognition*, pages 297–299. IEEE Computer Society Press, August 1998.

[3] Matej Artač, Matjaž Jogan, and Aleš Leonardis. Mobile robot localization using an incremental eigenspace model. In *ICRA 2002*. In Print, 2002.

[4] Dana Ballard. *Natural Computation.* The MIT Press, 1996.

[5] Horst Bischof, Horst Wildenauer, and Aleš Leonardis. Illumination insensitive eigenspaces. In *Proc. Intl. Conf. Computer Vision ICCV01*, pages I: 233–238. IEEE Computer Society, 2001.

[6] M. Dill, R. Wolf, and M. Heisenberg. Visual pattern recognition in *Drosophila* involves retinotopic matching. *Nature*, 365:751–753, 21 October 1993.

[7] José Gaspar, Niall Winters, and José Santos-Victor. Vision-based navigation and environmental representations with an omnidirectional camera. *IEEE Transaction on Robotics and Automation*, 16(6), December 2000.

[8] Chris Geyer and Kostas Daniilidis. Catadioptric projective geometry. *International Journal of Computer Vision*, (43):223–243, 2001.

[9] H. Ishiguro and S. Tsuji. Image-based memory of environment. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pages 634–639, 1996.

[10] Bernd Jähne, Horst Haußecker, and Peter Geißler, editors. *Handbook of Computer Vision and Applications*. Academic Press, 1999.

[11] Matjaž Jogan and Aleš Leonardis. Panoramic eigenimages for spatial localisation. In Franc Solina and Aleš Leonardis, editors, *8-th International Conference on Computer Analysis of Images and Patterns*, number 1689 in Lecture Notes in Computer Science, pages 558–567. Springer Verlag, September 1999.

[12] Matjaž Jogan and Aleš Leonardis. Robust localization using eigenspace of spinning-images. In *IEEE Workshop on Omnidirectional Vision*, pages 37–44. IEEE Computer Society, June 2000.

[13] Matjaž Jogan and Aleš Leonardis. Robust localization using panoramic view-based recognition. In *15th International Conference on Pattern Recognition*, volume 4, pages 136–139. IEEE Computer Society, September 2000.

[14] S.P.D. Judd and T.S. Collett. Multiple stored views and landmark guidance in ants. *Nature*, 392:710–714, 16 April 1998.

[15] Aleš Leonardis and Horst Bischof. Robust recognition using eigenimages. *Computer Vision and Image Understanding - Special Issue on Robust Statistical Techniques in Image Understanding*, 78(1):99–118, 2000.

[16] Aleš Leonardis, Horst Bischof, and Jasna Maver. Multiple eigenspaces. *Pattern Recognition*, 2002. In Press.

[17] Sakashi Maeda, Yoshinori Kuno, and Yoshiaki Shirai. Active navigation vision based on eigenspace analysis. In *Proc. Intl. Conf. on Intelligent Robots and Systems (IROS97)*, pages 1018–1023. IEEE Computer Society, 2001.

[18] Hiroshi Murase and Shree K. Nayar. Detection of 3D objects in cluttered scenes using hirearchical eigenspace. *Pattern recognition letters*, (18):375–384, 1997.

[19] D. Murray and C. Jennings. Stereo vision based mapping and navigation for mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation ICRA'97*, pages 1694–1699, April 1998.

[20] S. K. Nayar, S. A. Nene, and H. Murase. Subspace methods for robot vision. *IEEE Trans. on Robotics and Automation*, 12(5):750–758, October 1996.

[21] S.K. Nayar and T. Poggio, editors. *Early visual learning*. Oxford University Press, 1996.

[22] Tomáš Pajdla and Václav Hlaváč. Zero phase representation of panoramic images for image based localization. In Franc Solina and Aleš Leonardis, editors, *8-th International Conference on Computer Analysis of Images and Patterns*, number 1689 in Lecture Notes in Computer Science, pages 550–557. Springer Verlag, 1999.

[23] Lucas Paletta, Simone Frintropp, and Joachim Hertzberg. Robust localization using context in omnidirectional imaging. In *Proc. International Conference on Robotics and Automation, ICRA 2001*, pages 2072–2077, 2001.

[24] J. Ponce, A. Zisserman, and M. Hebert, editors. *Object Representation in Computer Vision II, ECCV'96 Int. Workshop, Cambridge*, volume 1114 of *Lecture Notes in Computer Science*. Springer, 1996.

[25] S. Se, D. Lowe, and J. Little. Vision-based mobile robot localization and mapping using scale-invariant features. In *Proceedings of the IEEE International Conference on Robotics and Automation ICRA'01*, pages 2051–2058, Seoul, Korea, May 2001.

[26] Gilbert Strang. The discrete cosine transform. *SIAM Review*, 41(1):135–147, 1999.

[27] Michihiro Uenohara and Takeo Kanade. Optimal approximation of uniformly rotated images: Relationships between Karhunen-Loeve expansion and discrete cosine transform. *IEEE Transactions on Image Processing*, 7(1):116–119, 1998.

[28] N. Vlassis, Y. Motomura, and B. Kröse. Supervised linear feature extraction for mobile robot localization. In B. Carlisle and O. Khatib, editors, *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 2979–2984. IEEE, ISBN 0-7803-5886-4/00, April 2000.

Izjava

Izjavljam, da sem diplomsko delo izdelal samostojno pod vodstvom mentorja doc. dr. Aleša Leonardisa. Izkazano pomoč drugih sodelavcev sem v celoti navedel v zahvali.